WEST

Help | Logout | Interrupt

Main Menu | Search Form | Posting Counts | Show S Numbers | Edit S Numbers | Preferences

Search Results -

Terms	Documents
(bus$3 or (sub adj1 bus$3)) same (bundle$1 or group$1 or together or package$1) same ((data adj1 flow adj1 processor$1) or (programmable adj1 gate adj1 array$1))	57

Database: US Patents Full-Text Database
JPO Abstracts Database
EPO Abstracts Database
Derwent World Patents Index
IBM Technical Disclosure Bulletins

Refine Search:

```
(bus$3 or (sub adj1 bus$3)) same
(bundle$1 or group$1 or together or
package$1) same ((data adj1 flow adj1
```

Clear

Search History

Today's Date: 11/30/2000

DB Name	Query	Hit Count	Set Name
USPT	(bus$3 or (sub adj1 bus$3)) same (bundle$1 or group$1 or together or package$1) same ((data adj1 flow adj1 processor$1) or (programmable adj1 gate adj1 array$1))	57	L1

WEST

Help | Logout | Interrupt

Main Menu | Search Form | Posting Counts | Show S Numbers | Edit S Numbers | Preferences

Search Results -

Terms	Documents
(bus$3 or (sub adj1 bus$3)) same (bundle$1 or group$1 or together or package$1) same ((data adj1 flow adj1 processor$1) or (programmable adj1 gate adj1 array$1))	2

Database: US Patents Full-Text Database; JPO Abstracts Database; EPO Abstracts Database; Derwent World Patents Index; IBM Technical Disclosure Bulletins

Refine Search: (bus$3 or (sub adj1 bus$3)) same (bundle$1 or group$1 or together or package$1) same ((data adj1 flow adj1

Clear

Search History

Today's Date: 11/30/2000

DB Name	Query	Hit Count	Set Name
JPAB	(bus$3 or (sub adj1 bus$3)) same (bundle$1 or group$1 or together or package$1) same ((data adj1 flow adj1 processor$1) or (programmable adj1 gate adj1 array$1))	2	L2
USPT	(bus$3 or (sub adj1 bus$3)) same (bundle$1 or group$1 or together or package$1) same ((data adj1 flow adj1 processor$1) or (programmable adj1 gate adj1 array$1))	57	L1

WEST

Help | Logout | Interrupt

Main Menu | Search Form | Posting Counts | Show S Numbers | Edit S Numbers | Preferences

Search Results -

Terms	Documents
(bus$3 or (sub adj1 bus$3)) same (bundle$1 or group$1 or together or package$1) same ((data adj1 flow adj1 processor$1) or (programmable adj1 gate adj1 array$1))	4

Database: US Patents Full-Text Database; JPO Abstracts Database; EPO Abstracts Database; Derwent World Patents Index; IBM Technical Disclosure Bulletins

Refine Search: (bus$3 or (sub adj1 bus$3)) same (bundle$1 or group$1 or together or package$1) same ((data adj1 flow adj1

Clear

Search History

Today's Date: 11/30/2000

DB Name	Query	Hit Count	Set Name
EPAB	(bus$3 or (sub adj1 bus$3)) same (bundle$1 or group$1 or together or package$1) same ((data adj1 flow adj1 processor$1) or (programmable adj1 gate adj1 array$1))	4	L3
JPAB	(bus$3 or (sub adj1 bus$3)) same (bundle$1 or group$1 or together or package$1) same ((data adj1 flow adj1 processor$1) or (programmable adj1 gate adj1 array$1))	2	L2
USPT	(bus$3 or (sub adj1 bus$3)) same (bundle$1 or group$1 or together or package$1) same ((data adj1 flow adj1 processor$1) or (programmable adj1 gate adj1 array$1))	57	L1

WEST

Help | Logout | Interrupt

Main Menu | Search Form | Posting Counts | Show S Numbers | Edit S Numbers | Preferences

Search Results -

Terms	Documents
(bus$3 or (sub adj1 bus$3)) same (bundle$1 or group$1 or together or package$1) same ((data adj1 flow adj1 processor$1) or (programmable adj1 gate adj1 array$1))	1

Database: US Patents Full-Text Database; JPO Abstracts Database; EPO Abstracts Database; Derwent World Patents Index; IBM Technical Disclosure Bulletins

Refine Search: (bus$3 or (sub adj1 bus$3)) same (bundle$1 or group$1 or together or package$1) same ((data adj1 flow adj1

Clear

Search History

Today's Date: 11/30/2000

DB Name	Query	Hit Count	Set Name
TDBD	(bus$3 or (sub adj1 bus$3)) same (bundle$1 or group$1 or together or package$1) same ((data adj1 flow adj1 processor$1) or (programmable adj1 gate adj1 array$1))	1	L4
EPAB	(bus$3 or (sub adj1 bus$3)) same (bundle$1 or group$1 or together or package$1) same ((data adj1 flow adj1 processor$1) or (programmable adj1 gate adj1 array$1))	4	L3
JPAB	(bus$3 or (sub adj1 bus$3)) same (bundle$1 or group$1 or together or package$1) same ((data adj1 flow adj1 processor$1) or (programmable adj1 gate adj1 array$1))	2	L2
USPT	(bus$3 or (sub adj1 bus$3)) same (bundle$1 or group$1 or together or package$1) same ((data adj1 flow adj1 processor$1) or (programmable adj1 gate adj1 array$1))	57	L1

WEST

Help | Logout | Interrupt

Main Menu | Search Form | Posting Counts | Show S Numbers | Edit S Numbers | Preferences

Search Results -

Terms	Documents
(710/100)!.CCLS. or 710/101.ccls. or 710/8.ccls. or 710/72.ccls. or 712/11.ccls. or 712/14.ccls. or 712/18.ccls. or 716/17.ccls. or 716/16.ccls. or 326/38.ccls. or 326/39.ccls. or 326/41.ccls.	3233

Database: US Patents Full Text Database
JPO Abstracts Database
EPO Abstracts Database
Derwent World Patents Index
IBM Technical Disclosure Bulletins

Refine Search: (710/100)!.CCLS. or 710/101.ccls. or 710/8.ccls. or 710/72.ccls. or 712/11.ccls. or 712/14.ccls. or

Clear

Search History

Today's Date: 11/30/2000

DB Name	Query	Hit Count	Set Name
USPT	(710/100)!.CCLS. or 710/101.ccls. or 710/8.ccls. or 710/72.ccls. or 712/11.ccls. or 712/14.ccls. or 712/18.ccls. or 716/17.ccls. or 716/16.ccls. or 326/38.ccls. or 326/39.ccls. or 326/41.ccls.	3233	L5
TDBD	(bus$3 or (sub adj1 bus$3)) same (bundle$1 or group$1 or together or package$1) same ((data adj1 flow adj1 processor$1) or (programmable adj1 gate adj1 array$1))	1	L4
EPAB	(bus$3 or (sub adj1 bus$3)) same (bundle$1 or group$1 or together or package$1) same ((data adj1 flow adj1 processor$1) or (programmable adj1 gate adj1 array$1))	4	L3
JPAB	(bus$3 or (sub adj1 bus$3)) same (bundle$1 or group$1 or together or package$1) same ((data adj1 flow adj1 processor$1) or (programmable adj1 gate adj1 array$1))	2	L2
USPT	(bus$3 or (sub adj1 bus$3)) same (bundle$1 or group$1 or together or package$1) same ((data adj1 flow adj1 processor$1) or (programmable adj1 gate adj1 array$1))	57	L1

WEST

Help | Logout | Interrupt

Main Menu | Search Form | Posting Counts | Show S Numbers | Edit S Numbers | Preferences

Search Results -

Terms	Documents
l1 and l5	17

Database: US Patents Full-Text Database; JPO Abstracts Database; EPO Abstracts Database; Derwent World Patents Index; IBM Technical Disclosure Bulletins

Refine Search: `l1 and l5` Clear

Search History

Today's Date: 11/30/2000

DB Name	Query	Hit Count	Set Name
USPT	l1 and l5	17	L6
USPT	(710/100)!.CCLS. or 710/101.ccls. or 710/8.ccls. or 710/72.ccls. or 712/11.ccls. or 712/14.ccls. or 712/18.ccls. or 716/17.ccls. or 716/16.ccls. or 326/38.ccls. or 326/39.ccls. or 326/41.ccls.	3233	L5
TDBD	(bus$3 or (sub adj1 bus$3)) same (bundle$1 or group$1 or together or package$1) same ((data adj1 flow adj1 processor$1) or (programmable adj1 gate adj1 array$1))	1	L4
EPAB	(bus$3 or (sub adj1 bus$3)) same (bundle$1 or group$1 or together or package$1) same ((data adj1 flow adj1 processor$1) or (programmable adj1 gate adj1 array$1))	4	L3
JPAB	(bus$3 or (sub adj1 bus$3)) same (bundle$1 or group$1 or together or package$1) same ((data adj1 flow adj1 processor$1) or (programmable adj1 gate adj1 array$1))	2	L2
USPT	(bus$3 or (sub adj1 bus$3)) same (bundle$1 or group$1 or together or package$1) same ((data adj1 flow adj1 processor$1) or (programmable adj1 gate adj1 array$1))	57	L1

WEST

Help | Logout | Interrupt

Main Menu | Search Form | Posting Counts | Show S Numbers | Edit S Numbers | Preferences

Search Results -

Terms	Documents
l1 and l5	17

Database: US Patents Full-Text Database
JPO Abstracts Database
EPO Abstracts Database
Derwent World Patents Index
IBM Technical Disclosure Bulletins

Refine Search: `l1 and l5` Clear

Search History

Today's Date: 11/30/2000

DB Name	Query	Hit Count	Set Name
USPT	l1 and l5	17	L6
USPT	(710/100)!.CCLS. or 710/101.ccls. or 710/8.ccls. or 710/72.ccls. or 712/11.ccls. or 712/14.ccls. or 712/18.ccls. or 716/17.ccls. or 716/16.ccls. or 326/38.ccls. or 326/39.ccls. or 326/41.ccls.	3233	L5
TDBD	(bus$3 or (sub adj1 bus$3)) same (bundle$1 or group$1 or together or package$1) same ((data adj1 flow adj1 processor$1) or (programmable adj1 gate adj1 array$1))	1	L4
EPAB	(bus$3 or (sub adj1 bus$3)) same (bundle$1 or group$1 or together or package$1) same ((data adj1 flow adj1 processor$1) or (programmable adj1 gate adj1 array$1))	4	L3
JPAB	(bus$3 or (sub adj1 bus$3)) same (bundle$1 or group$1 or together or package$1) same ((data adj1 flow adj1 processor$1) or (programmable adj1 gate adj1 array$1))	2	L2
USPT	(bus$3 or (sub adj1 bus$3)) same (bundle$1 or group$1 or together or package$1) same ((data adj1 flow adj1 processor$1) or (programmable adj1 gate adj1 array$1))	57	L1

Your search matched **2** of **645966** documents.
Results are shown **15** to a page, sorted by **publication year** in **descending** order.
You may refine your search by editing the current search expression or entering a new one the text box.
Then click **Search Again**.

((bus?)and ((bundle? or group?))) and((processor? Search Again

Results:
Journal or Magazine = **JNL** Conference = **CNF** Standard = **STD**

1 **Exploiting inter task dependencies for dynamic load balancing**
Becker, W.; Waldmann, G.
High Performance Distributed Computing, 1994., Proceedings of the Third IEEE International Sym 1994
Page(s): 157 -165

[Abstract] [PDF Full-Text] **CNF**

2 **Scan line graphics generation on the massively parallel processor**
Dorband, J.E.
Frontiers of Massively Parallel Computation, 1988. Proceedings., 2nd Symposium on the Frontiers
Page(s): 327 -329

[Abstract] [PDF Full-Text] **CNF**

Exploiting inter task dependencies for dynamic load balancing

- Becker, W.; Waldmann, G.
Inst. for Parallel & Distributed High Performance Syst., Stuttgart Univ., Germany

This paper appears in: **High Performance Distributed Computing, 1994., Proceedings of the Th International Symposium on**
On page(s): 157 - 165
2-5 Aug. 1994
1994
ISBN: 0-8186-6395-2
IEEE Catalog Number: 94TH0667-6
Number of Pages: xiii+304
References Cited: 21
INSPEC Accession Number: 4778644

Abstract:
The major goal of dynamic load balancing is not primarily to equalize the load on the nodes of a paral system, but to optimize the average response time of single requests or the throughput of all applicatio system. Therefore it is often necessary not only to keep all processors busy and all processor ready qu within the same range, but to avoid delays and inefficient computations caused by foreseeable but igno and precedence constraints between related tasks. We present concepts for dynamic consideration of i dependencies within small groups of tasks and evaluate them observing real applications in a load bal environment on a network of workstations. The concepts are developed from scheduling of single task towards heterogeneous multi user operation scenarios.

Index Terms:
resource allocation; scheduling; optimisation; computer networks; workstations; inter task dependenc load balancing; parallel computing system; response time; optimization; processor ready queue length inefficient computation; data flow; workstation network; scheduling; single task graphs; heterogeneou operation scenarios

SEARCH RESULTS PDF FULL-TEXT PREVIOUS

Scan line graphics generation on the massively parallel processor

- *Dorband, J.E.*
NASA, Greenbelt, MD, USA

This paper appears in: **Frontiers of Massively Parallel Computation, 1988. Proceedings., 2nd S the Frontiers of**
On page(s): 327 - 329
10-12 Oct. 1988
1989
ISBN: 0-8186-5892-4
Number of Pages: xxxxi+736
References Cited: 3
INSPEC Accession Number: 3532276

Abstract:
The author describes the implementation of a scan line graphics generation algorithm on the massivel processor MPP. Pixels are computed in parallel and their results are applied to the Z buffer in large gr perform pixel value calculations, facilitate load balancing across the processors and apply the results t efficiently in parallel requires special virtue routing (sort computation) techniques developed by the a especially for use on single-instruction multiple-data (SIMD) architectures. This involves a preproces step which determines how much of the sort is necessary to provide sufficient contiguous space to dup data. Once this has been determined a sort is used to compress the data which can be terminated early information derived by the scout step. This then gives the ability to keep as many processors as possib reasonable efficiency.

Index Terms:
pixel brightness; massively parallel processor; scan line graphics generation algorithm; Z buffer; pixe calculations; load balancing; virtue routing; sort computation; single-instruction multiple-data; SIMD graphics; parallel algorithms; parallel architectures

SEARCH RESULTS PDF FULL-TEXT PREVIOUS

Home | Log-out | Journals | Conference Proceedings | Standards
Search by Author | Basic Search | Advanced Search | Join IEEE | Establish a Web Account

WEST

End of Result Set

Generate Collection

L2: Entry 2 of 2 File: JPAB Aug 7, 1985

DOCUMENT-IDENTIFIER: JP 60150139 A
TITLE: DATA FLOW PROCESSOR

FPAR:
PURPOSE: To attain the parallel processing for transfer of data and to increase the processing speed of a data flow processor, by using a detecting function for data transfer to plural bus groups as well as a simultaneous access function for memory addresses of instructions.

FPAR:
CONSTITUTION: The instructions stored in a memory part M are sent to the 1st bus groups Bus1-1∼Bus3-1 via an interface part Di together with data and then undergo the airthmetic processing through corresponding arithmetic parts f1∼f3 via an interface R. The results of the arithmetic processing are delivered after the idle state is decided at an arithmetic output interface part D for the 2nd bus groups Bus1-2∼Bus3-2. Then an interface part A of the part M decodes the address information to which the next execution instruction is designated for the data on the 2nd bus groups when these groups are filled. Then an access is given to an instruction corresponding to the decoded information from the part M. This instruction is sent to the 1st bus together with the next arithmetic object data through the part Di. Then the transfer of data is processed in parallel. This can accelerate the processing speed of a data flow processor.

WEST

End of Result Set

Generate Collection

L4: Entry 1 of 1 File: TDBD Mar 1, 1980

DOCUMENT-IDENTIFIER: NN80034369
TITLE: Storage Protection Mechanism for Processor. March 1980.

TBTX:
3p. This article describes a mechanism for checking for possible software errors that could disrupt the integrity of the main storage in a processor, thus providing some degree of storage protection. The Storage Protection Device utilizes circuitry shown in Fig. 1 and is adaptable to a system without affecting the hardware or data flow of a processor having a Storage Relocation Translator such as that described in U.S. Patents 4,037,215 and 4,050,094. It serves to control processor channel functions without being part of the channel. - Storage access is controlled through bits contained in the storage relocation translator segmentation registers. The segmentation register bit format is shown in Fig. 2. - Bit S (Summary)- This bit if set to "1" indicates that the programmer has set his address to point to a location greater than the maximum storage defined for that processor. Therefore, the hardware will prevent this storage access and flag the address as an Invalid Storage Address (ISA). - Bits A, 0, 1, 2, 3, 4 -These are the high-order physical address bits of a 17 bit address. - Bit V (Valid) - This bit must be set to a "1". If set to zero the address associated with that segmentation register is considered invalid. If an attempt is made to use this address, the hardware will prevent this storage access and flag the address as an ISA. - Bit R (Read Only) -- If set to a "1", it serves to block storage writes to the location in storage specified by that segmentation register. An attempt to write to this location with the read-only bit "on" will result in a Protect Check in the processor status word (PSW) and an end to the storage access. - A storage cycle is initiated by a Set Storage Address Register instruction. This is followed by a signal from the channel to storage, called Storage Gate A, which indicates that a storage address is on the Storage Address Bus. This is followed by another signal from the channel to storage called Storage Gate E, which gates data to the Data Bus. When Storage Gate A and Storage Gate E are received by storage, a signal from storage to the channel is generated called Synchronous Storage Return, meaning that storage is capable of operating at synchronous speed. Should this cycle not be completed within a predetermined time interval such as 6.4 microseconds, the channel will time out and the processor will generate an ISA check condition. - The circuitry of Fig. 1 checks for the invalid conditions aforementioned and determines whether Storage Gate B will be blocked or not. If Storage Gate B is blocked, the storage access cycle will not be complete. Synchronous Storage Return would never be generated, and the channel would time out. - The logic circuitry in Fig. 1 is active only when the storage relocation translator is enabled, as evidenced by active control lines 1 and 2. The upper group of circuits 3-5 provide the checking operation when the storage transfer is either between the processor and main storage or between the channel and main storage when operating in a direct program control (DPC) mode. Control line 6 disables the Storage Gate B line to gate 7 when segmentation register bit V indicates a not-valid condition. Control line 8 will disable the gate 7 when bit S indicates a programming error when initially loading the segmentation registers. The output of AND circuit 5 will disable the gate 7 when the processor is doing a write to main storage and bit R indicates a read-only status. - The lower group of circuits 10-13 perform a similar checking operation for segmentation register bits S and V when cycle steal data transfers are being performed between the channel and the main storage unit. Gate 7 is disabled if bit S has a value of 1 (programming error) or bit V has a value of 0 (not valid). Circuits 12 and 13 provide a latch circuit, line 14 being the set Input and line 15 being the reset input.

WEST

End of Result Set

Generate Collection

L1: Entry 2 of 2 File: USPT Aug 24, 1999

US-PAT-NO: 5943242
DOCUMENT-IDENTIFIER: US 5943242 A

TITLE: Dynamically reconfigurable data processing system

DATE-ISSUED: August 24, 1999

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Vorbach; Martin Andreas	Karlsruhe	N/A	N/A	DEX
Munch; Robert Markus	Karlsruhe	N/A	N/A	DEX

ASSIGNEE INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY	TYPE CODE
PACT GmbH	Karlsruher	N/A	N/A	DEX	03

APPL-NO: 8/ 544435
DATE FILED: November 17, 1995

INT-CL: [6] G06F 13/00, G06F 17/00
US-CL-ISSUED: 364/491; 395/800.11, 395/800.14, 395/800.18
US-CL-CURRENT: 716/17; 712/11, 712/14, 712/18
FIELD-OF-SEARCH: 364/488-491, 364/749, 364/745, 364/716.02, 364/716.03, 364/716.01, 364/716.05, 395/800, 395/707, 395/800.16, 395/800.14, 395/800.15, 395/800.18, 395/800.11

REF-CITED:

U.S. PATENT DOCUMENTS

Search Selected Search ALL

	PAT-NO	ISSUE-DATE	PATENTEE-NAME	US-CL
☐	Re34363	August 1993	Freeman	326/38
☐	4706216	November 1987	Carter	365/94
☐	4739474	April 1988	Holsztynski et al.	395/800.14
☐	4761755	August 1988	Ardini, Jr. et al.	395/800.27
☐	4811214	March 1989	Nosenchuck et al.	395/800.11
☐	4870302	September 1989	Freeman	326/41
☐	4901268	February 1990	Judd	364/748.19
☐	4967340	October 1990	Dawes	395/800.19
☐	5014193	May 1991	Garner et al.	395/830
☐	5015884	May 1991	Agrawal et al.	326/39

5021947	June 1991	Campbell et al.	395/800.25
5023775	June 1991	Poret	326/40
5081375	January 1992	Pickett et al.	70/472
5109503	April 1992	Cruickshank et al.	395/500
5123109	June 1992	Hillis	395/800.22
5125801	June 1992	Nabity et al.	417/44.1
5128559	July 1992	Steele	326/38
5142469	August 1992	Weisenborn	364/146
5204935	April 1993	Mihara et al.	395/3
5208491	May 1993	Ebeling et al.	326/41
5226122	July 1993	Thayer et al.	395/500
5233539	August 1993	Agrawal et al.	364/489
5247689	September 1993	Ewert	395/311
5287472	February 1994	Horst	365/189.08
5301344	April 1994	Kolchinsky	395/800.32
5303172	April 1994	Magar et al.	364/726.04
5336950	August 1994	Popli et al.	326/39
5361373	November 1994	Gilson	395/800.01
5418952	May 1995	Morley et al.	395/800.14
5421019	May 1995	Holsztynski et al.	395/800.14
5422823	June 1995	Agrawal et al.	364/489
5426378	June 1995	Ong	326/39
5430687	July 1995	Hung et al.	364/230.08
5440245	August 1995	Galbraith et al.	326/38
5442790	August 1995	Nosenchuck	395/707
5444394	August 1995	Watson et al.	326/45
5448186	September 1995	Kawata	326/41
5455525	October 1995	Ho et al.	326/41
5457644	October 1995	McCollum	364/716.01
5473266	December 1995	Ahanin et al.	326/41
5473267	December 1995	Stansfield	326/41
5475583	December 1995	Bock et al.	364/141
5475803	December 1995	Stearns et al.	395/136
5483620	January 1996	Pechanek et al.	395/27
5485103	January 1996	Pedersen et al.	326/41
5485104	January 1996	Agrawal et al.	326/38
5489857	February 1996	Agrawal et al.	326/41
5491353	February 1996	Kean	257/208

5493239	February 1996	Zlotnick	326/38
5497498	March 1996	Taylor	395/284
5506998	April 1996	Kato et al.	395/800.29
5510730	April 1996	El Gamal et al.	326/41
5511173	April 1996	Yamaura et al.	395/598
5513366	April 1996	Agarwal et al.	395/800.22
5521837	May 1996	Frankle et al.	364/491
5522083	May 1996	Gove et al.	395/800.22
5532693	July 1996	Winters et al.	341/51
5532957	July 1996	Malhi	365/154
5535406	July 1996	Kolchinsky	395/800.1
5537057	July 1996	Leong et al.	326/41
5537601	July 1996	Kimura et al.	395/800.35
5541530	July 1996	Cliff et al.	326/41
5544336	August 1996	Kato et al.	395/311
5548773	August 1996	Kemeny et al.	395/800.11
5555434	September 1996	Carlstedt	395/800.38
5559450	September 1996	Ngai et al.	326/40
5561738	October 1996	Kinerk et al.	395/3
5570040	October 1996	Lytle et al.	326/41
5583450	December 1996	Trimberger et al.	326/41
5586044	December 1996	Agrawal et al.	364/489
5587921	December 1996	Agrawal et al.	364/489
5588152	December 1996	Dapp et al.	395/800.16
5590345	December 1996	Barker et al.	395/800.11

FOREIGN PATENT DOCUMENTS

FOREIGN-PAT-NO	PUBN-DATE	COUNTRY
0 221 360	May 1987	EPX
0428327A1	November 1990	EPX
0539595A1	April 1992	EPX
0 678 985	October 1995	EPX
0 726 532	August 1996	EPX
WO90/11648	October 1990	WOX
94/08399	April 1994	WOX

OTHER PUBLICATIONS

M. Saleeba, "A Self-Contained Dynamically Reconfigurable Processor Architecture", Sixteenth Australian Computer Science Conference, ASCS-16, Brisbane, QLD., Australia, Feb., 1993.
Villasenor, John, et al., "Configurable Computing," Scientific American, vol. 276, No. 6, Jun. 1997, pp. 66-71.
Villasenor, John, et al., "Configurable Computing Solutions for Automatic Target

Recognition," IEEE, 1996, pp. 70-79.
Athanas, Peter, et al., "IEEE Symposium on FPGAs For Custom Computing Machines," IEEE Computer Society Press, Apr. 19-21, 1995, pp. i-vii, 1-122.
Bittner, Ray, A., Jr., "Wormhole Run-Time Reconfiguration: Conceptualization and VLSI Design of a High Performance Computing system," Dissertation, Jan. 23, 1997, pp. i-xx, 1-415.
Myers, G. , Advances in Computer Architecture, Wiley-Interscience Publication, 2nd ed., John Wiley & Sons, Inc., pp. 463-494, 1978.
M. Morris Mano "Digital Design," by Prentic-Hall, Inc., Englewood Cliffs, New Jersey 07632, 1984, pp. 119-125, 154-161.

ART-UNIT: 273
PRIMARY-EXAMINER: Teska; Kevin J.
ASSISTANT-EXAMINER: Siek; Vuthe
ATTY-AGENT-FIRM: Kenyon & Kenyon

ABSTRACT:

A data processing system, wherein a data flow processor (DFP) integrated circuit chip is provided which comprises a plurality of orthogonally arranged homogeneously structured cells, each cell having a plurality of logically same and structurally identically arranged modules. The cells are combined and facultatively grouped using lines and columns and connected to the input/output ports of the DFP. A compiler programs and configures the cells, each by itself and facultatively grouped, such that random logic functions and/or linkages among the cells can be realized. The manipulation of the DFP configuration is performed during DFP operation such that modification of function parts (MACROs) of the DFP can take place without requiring other function parts to be deactivated or being impaired.

23 Claims, 30 Drawing figures

WEST

End of Result Set

☐ Generate Collection

L1: Entry 2 of 2 File: USPT Aug 24, 1999

US-PAT-NO: 5943242
DOCUMENT-IDENTIFIER: US 5943242 A

TITLE: Dynamically reconfigurable data processing system

DATE-ISSUED: August 24, 1999

INT-CL: [6] G06F 13/00, G06F 17/00
US-CL-ISSUED: 364/491; 395/800.11, 395/800.14, 395/800.18
US-CL-CURRENT: 716/17; 712/11, 712/14, 712/18
FIELD-OF-SEARCH: 364/488-491, 364/749, 364/745, 364/716.02, 364/716.03, 364/716.01, 364/716.05, 395/800, 395/707, 395/800.16, 395/800.14, 395/800.15, 395/800.18, 395/800.11

WEST

Generate Collection

L1: Entry 1 of 2 File: USPT Sep 12, 2000

US-PAT-NO: 6119181
DOCUMENT-IDENTIFIER: US 6119181 A

TITLE: I/O and memory bus system for DFPs and units with two- or multi-dimensional programmable cell architectures

DATE-ISSUED: September 12, 2000

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Vorbach; Martin	Karlsruhe	N/A	N/A	DEX
Munch; Robert	Karlsruhe	N/A	N/A	DEX

ASSIGNEE INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY	TYPE CODE
PACT GmbH	Munich	N/A	N/A	DEX	03

APPL-NO: 8/ 947254
DATE FILED: October 8, 1997

FOREIGN-APPL-PRIORITY-DATA:

COUNTRY	APPL-NO	APPL-DATE
DE	196 54 595	December 20, 1996

INT-CL: [7] G06F 13/40, G06F 15/80
US-CL-ISSUED: 710/100; 710/110, 710/129, 712/11
US-CL-CURRENT: 710/100; 710/110, 710/129, 712/11
FIELD-OF-SEARCH: 395/280, 395/309, 395/800.11, 395/80.25, 395/800.29, 395/377, 326/38, 326/39, 326/41, 364/489, 370/419, 710/100, 710/129, 710/110, 712/11, 712/14, 712/18, 712/25, 712/29, 712/31, 711/100

REF-CITED:

U.S. PATENT DOCUMENTS

Search Selected Search ALL

	PAT-NO	ISSUE-DATE	PATENTEE-NAME	US-CL
☐	Re34363	August 1993	Freeman	326/38
☐	4706216	November 1987	Carter	365/94
☐	4739474	April 1988	Holsztynski et al.	395/800.14
☐	4761755	August 1988	Ardini et al.	364/749
☐	4811214	March 1989	Nosenchuck et al.	395/800.11
☐	4852048	July 1989	Morton	712/11
☐	4870302	September 1989	Freeman	326/41
☐	4901268	February 1990	Judd	364/748.19

☐	4967340	October 1990	Dawes	395/800.19
☐	5014193	May 1991	Garner et al.	395/830
☐	5015884	May 1991	Agrawal et al.	326/39
☐	5021947	June 1991	Campbell et al.	395/800.25
☐	5023775	June 1991	Poret	326/40
☐	5081375	January 1992	Pickett et al.	326/41
☐	5109503	April 1992	Cruickshank et al.	395/500
☐	5113498	May 1992	Evan et al.	395/828
☐	5123109	June 1992	Hillis	395/800.22
☐	5125801	June 1992	Nabity et al.	417/44.1
☐	5128559	July 1992	Steele	326/38
☐	5142469	August 1992	Weisenborn	364/146
☐	5204935	April 1993	Mihara et al.	706/4
☐	5208491	May 1993	Ebeling et al.	326/41
☐	5226122	July 1993	Thayer et al.	395/500
☐	5233539	August 1993	Agrawal et al.	364/489
☐	5247689	September 1993	Ewert	395/311
☐	5287472	February 1994	Horst	365/189.08
☐	5301344	April 1994	Kolchinsky	395/800.32
☐	5303172	April 1994	Magar et al.	364/726.04
☐	5336950	August 1994	Popli et al.	326/39
☐	5361373	November 1994	Gilson	395/800.01
☐	5418952	May 1995	Morley et al.	395/800.14
☐	5421019	May 1995	Holsztynski et al.	395/800.14
☐	5422823	June 1995	Agrawal et al.	364/489
☐	5426378	June 1995	Ong	326/39
☐	5430687	July 1995	Hung et al.	365/230.08
☐	5440245	August 1995	Galbraith et al.	326/38
☐	5442790	August 1995	Nosenchuck	395/707
☐	5444394	August 1995	Watson et al.	326/45
☐	5448186	September 1995	Kawata	326/41
☐	5455525	October 1995	Ho et al.	326/41
☐	5457644	October 1995	McCollum	364/716.01
☐	5473266	December 1995	Ahanin et al.	326/41
☐	5473267	December 1995	Stansfield	326/41
☐	5475583	December 1995	Bock et al.	364/141
☐	5475803	December 1995	Stearns et al.	345/436
☐	5483620	January 1996	Pechanek et al.	706/41

☐	5485103	January 1996	Pedersen et al.	326/41
☐	5485104	January 1996	Agrawal et al.	326/38
☐	5489857	February 1996	Agrawal et al.	326/41
☐	5491353	February 1996	Kean	257/208
☐	5493239	February 1996	Zlotnick	326/38
☐	5497498	March 1996	Taylor	395/284
☐	5506998	April 1996	Kato et al.	395/800.29
☐	5510730	April 1996	El Gamal et al.	326/41
☐	5511173	April 1996	Yamaura et al.	395/598
☐	5513366	April 1996	Agarwal et al.	395/800.22
☐	5521837	May 1996	Frankle et al.	364/491
☐	5522083	May 1996	Gove et al.	395/800.22
☐	5532693	July 1996	Winters et al.	341/51
☐	5532957	July 1996	Malhi	365/154
☐	5535406	July 1996	Kolchinsky	395/800.1
☐	5537057	July 1996	Leong et al.	326/41
☐	5537601	July 1996	Kimura et al.	395/800.35
☐	5541530	July 1996	Cliff et al.	326/41
☐	5544336	August 1996	Kato et al.	395/311
☐	5548773	August 1996	Kemney et al.	395/800.11
☐	5555434	September 1996	Carlstedt	395/800.33
☐	5559450	September 1996	Ngai et al.	326/40
☐	5561738	October 1996	Kinerk et al.	706/4
☐	5570040	October 1996	Lytle et al.	326/41
☐	5583450	December 1996	Trimberger et al.	326/41
☐	5586044	December 1996	Agrawal et al.	364/489
☐	5587921	December 1996	Agrawal et al.	364/489
☐	5588152	December 1996	Dapp et al.	395/800.16
☐	5590345	December 1996	Barker et al.	395/800.11
☐	5659797	August 1997	Zandveld et al.	710/22
☐	5844888	December 1998	Markkula, Jr. et al.	370/255

FOREIGN PATENT DOCUMENTS

FOREIGN-PAT-NO	PUBN-DATE	COUNTRY
0 221 360	May 1987	EPX
0428327A1	May 1991	EPX
748 051 A2	December 1991	EPX
0539595A1	May 1993	EPX
0 678 985	October 1995	EPX
0 726 532	August 1996	EPX
735 685	October 1996	EPX
4416881	November 1994	DEX
WO90/11648	October 1990	WOX
94/08399	April 1994	WOX

OTHER PUBLICATIONS

Villasenor, John, et al., "Configurable Computing." Scientific American, vol. 276, No. 6, Jun. 1997, pp. 66-71.
Villasenor, John, et al., "Configurable Computing Solutions for Automatic Target Recognition," IEEE, 1996 pp. 70-79.
Tau Edward, et al., "A First Generation DPGA Implementation," FPD'95, pp. 138-143.

Athanas, Peter, et al. "IEEE Symposium of FPGAs For Custom Computing Machines," IEEE Computer Society Press, Apr. 19-21, 1995, pp. i-vii, 1-222.
Bittner, Ray, A., Jr., "Wormhole Run-Time Reconfiguration: Conceptualization and VLSI Design of a High Performance Computing system," Dissertation, Jan. 23, 1997, pp. i-xx, 1-415.
Meyers, G., Advances in Computer Architecture, Wiley-Interscience Publication, 2nd ed., John Wiley & Sons, Inc. pp. 463-494, 1978.
M. Saleeba, "A Self-Contained Dynamically Reconfigurable Processor Architecture", Sixteenth Australian Computer Science Conference, ASCS-16 QLD, Australia, Feb., 1993.
M. Moriis Mano, "Digital Design," by Prentice Hall, Inc., Englewood Cliffs, New Jersey 07632, 1984, pp. 119-125, 154-161.
Maxfield C. "Logic that Mutates While-U-Wait" EDN (Bur. Ed) (USA), EDN (European Edition), Nov. 7, 1996, Cahners Publishing, USA.

ART-UNIT: 271
PRIMARY-EXAMINER: Ray; Gopal C.
ATTY-AGENT-FIRM: Kenyon & Kenyon

ABSTRACT:

A uniform bus system is provided which operates without any special consideration by a programmer. Memories and peripheral may be connected to this bus system without any special measures. Likewise, units may be cascaded with the help of the bus system. The bus system combines a number of internal lines, and leads them as a bundle to terminals. The bus system control is predefined and does not require any influence by the programmer. Any number of memories, peripherals or other units can be connected to the bus system.

35 Claims, 20 Drawing figures

WEST

Generate Collection

L1: Entry 1 of 2 File: USPT Sep 12, 2000

US-PAT-NO: 6119181
DOCUMENT-IDENTIFIER: US 6119181 A

TITLE: I/O and memory bus system for DFPs and units with two- or multi-dimensional programmable cell architectures

DATE-ISSUED: September 12, 2000

INT-CL: [7] G06F 13/40, G06F 15/80
US-CL-ISSUED: 710/100; 710/110, 710/129, 712/11
US-CL-CURRENT: 710/100; 710/110, 710/129, 712/11
FIELD-OF-SEARCH: 395/280, 395/309, 395/800.11, 395/80.25, 395/800.29, 395/377, 326/38, 326/39, 326/41, 364/489, 370/419, 710/100, 710/129, 710/110, 712/11, 712/14, 712/18, 712/25, 712/29, 712/31, 711/100